AD-753 702

LOW-COST INTERACTIVE IMAGE PROCESSING

Emily G. Johnston, et al

Maryland University

Prepared for:

Air Force Office of Scientific Research

October 1972

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

AFOSR - TR - 72 - 2330

LOW-COST INTERACTIVE IMAGE PROCE

Emily G. Johnston, et al

Maryland University

Approved for public release;
distribution unlimited.

UNIVERSITY OF MARYLAND

COMPUTER SCIENCE CENTER

COLLEGE PARK, MARYLAND

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) University of Maryland Computer Science Center College Park, Maryland 20742	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

LOW-COST INTERACTIVE IMAGE PROCESSING

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Scientific Interim

5. AUTHOR(S) (First name, middle initial, last name)

Emily G. Johnston
Azriel Rosenfeld

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
October 1972	24	5

8a. CONTRACT OR GRANT NO. F44620-72-C-0062 b. PROJECT NO. 9769 c. 61102F d. 681304	9a. ORIGINATOR'S REPORT NUMBER(S) TR-202 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFOSR - TR - 72 - 2330

10. DISTRIBUTION STATEMENT

A. Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES TECH OTHER	12. SPONSORING MILITARY ACTIVITY Air Force Office of Scientific Rsch (NM) 1400 Wilson Blvd Arlington, Virginia 22209

13. ABSTRACT

This paper describes how to do useful, nontrivial image processing tasks interactively using only a standard alphanumeric CRT terminal, or even a teletype. Only an ordinary time-sharing system is required; there is no need for a dedicated computer or channel, or even for special priority on the system.

I

DD FORM 1 NOV 65 1473

Security Classification

Technical Report TR-202 October, 1972
F44620-72-C-0062

LOW-COST INTERACTIVE IMAGE PROCESSING

Emily G. Johnston
Azriel Rosenfeld

ABSTRACT

This paper describes how to do useful, nontrivial image processing tasks interactively using only a standard alphanumeric CRT terminal, or even a teletype. Only an ordinary time-sharing system is required; there is no need for a dedicated computer or channel, or even for special priority on the system.

The support of the Air Force Office of Scientific Research under Contract F44620-72-C-0062 is gratefully acknowledged, as is the help of Eleanor B. Waters in preparing this paper.

Introduction

The advantages of interactive processing over batch processing are by now well established. When working in an interactive mode, the programmer obtains quick responses to his actions, and does not have to reconstruct his line of reasoning each time a response is obtained. This is especially important when the data being processed are graphical or pictorial; it is of great benefit to be able to see the results of each processing step displayed immediately. In addition, it is crucial to be able to point to objects in the display, or outline regions in the display, in such a way that the computer knows which objects or regions are intended; this ability is impossibly cumbersome to achieve in batch mode.

Low-cost interactive graphics terminals are now widely available, but one hears much less about interactive image processing systems. Of the available computer image displays, only the more expensive permit any sort of interaction, such as pointing, outlining, or selective modification.

This paper describes an approach to interactive image processing using only a standard alphanumeric CRT terminal--or if necessary, even a teletype. This approach can be implemented on any ordinary time-sharing system; it does not require a dedicated computer, a dedicated channel, or even special priority on the system.

Input

Digitized images, even of moderate size, contain enormous numbers of bits; for example, a commercial TV picture contains about 500 by 500 resolvable points, and if we represent the gray level of each point by a 6-bit number, we have $1\frac{1}{2}$ million bits in the picture. For this reason, it has often been suggested that, when doing image processing on a computer, one should not digitize the entire image and input it to the computer; rather, one should allow the computer to control a scanning device which can read and digitize selected portions of the image on demand. If this suggestion is accepted, it implies a major hardware expense before the image processing itself can even begin.

Fortunately, there are alternatives to the computer-controlled scanner approach, provided that one is willing to trade I/O time for hardware cost. A digitized TV picture occupies only a few yards of magnetic tape; one can store many such pictures on a single tape. If desired, the picture can be transferred to disk or drum storage before beginning its actual processing. I/O limitations may make it somewhat slower to access pictures from disk than to access them by controlling a scanner, but this is a modest price to pay in place of having to buy or build the scanner.

There are also counterarguments to the scanner concept itself. Information obtained by the scanner may not be perfectly repeatable,

since analog signals are involved; in some circumstances, this may be highly objectionable. In many cases, too, the processing to be done on the input image requires many times more storage than does that image itself, so that the storage saving resulting from being able to use the image as a memory is negligible.*

Digitized images on magnetic tape are widely available from many sources, and digitizing services are also available, at a cost of only a few dollars per image. Thus anyone wishing to undertake experimental image processing can provide himself with a data base very cheaply. Experience suggests, moreover, that most experimenters will need only a few digital images for program development and feasibility testing. In the early stages of image processing research, one usually wants to try out many techniques on the same image, not one technique on many images.

One can even do non-trivial image processing research on computer-created images (e.g., defined by stochastic processes).

In fact, it is wise to develop and test one's image processing software using simple test-pattern images, so that there is no doubt what the results of the operations are supposed to look like.

---

*The processing is inexpensive in terms of storage only when it involves relatively rare events that can be detected in the image as it is scanned, so that most of the image need not be stored.

Output

When processing images interactively, one usually need not look at the entire image after each step. Display of small pieces should generally be adequate for checking that the correct image has been read in, or that a step has been executed correctly. An example is shown in Figure 1; it is a 72-by-44-point image, where each point has one of 32 shades of gray.

An output such as that in Figure 1 requires grayscale display hardware that is not normally available to an ordinary time-sharing system user. Moreover, the time required to output Figure 1, say on a 300-baud line, would be about two minutes (assuming one character per image point). Given a display without grayscale capability, if the user attempted to achieve grayscale by exciting display points repeatedly, the time would become even greater. Moreover, the resulting display is small (Figure 1 already involves considerable defocus), making it hard to interact with the displayed image accurately.

Most of these difficulties can be avoided by using arrays of alphanumeric characters for image output. One now needs only an ordinary alphanumeric terminal (even a teletype will do), though of course a graphics terminal too can be used in this way. Gray shades are represented using characters that have various ratios of character area to background area, ranging from blank to (say) W. One can construct an adequate 8-step

y scale in this way with single characters; and if overstrike is permitted, a 32-step scale can be obtained. Figure 2 shows the image of Figure 1 output in this way, and the character sets used. The overstriking is achieved by eliminating the line feed sent by the computer. Unfortunately, one cannot do "overstriking" on a CRT terminal, but it can be done on the CRT's auxiliary hard-copy printer. There is a vertical/horizontal scale distortion of about 3:2, but this is not objectionable for most purposes. (One can, of course, also use non-gray scale alphanumeric output (e.g., gray levels 0,..,31 = blank, 1,...,9, A,...,V) if one wishes to read gray levels of individual points rather than see the points as an image; see Figure 2e.)

The overstruck gray scale is much better than the single-character scale, but it has the disadvantage of being far more time consuming. Outputting a 72-by-40 single-character image on a CRT terminal operating at 1200 baud takes less than 30 seconds; but outputting an overstruck image of the same size on a teletype at 110 baud takes nearly ten minutes, which is impractically tedious.

If the single-character scheme is used, it is important to pick the gray level ranges that correspond to the characters carefully. A good rule of thumb is to pick the ranges to contain numbers of picture points that are as equal as possible. The 8-level grayscale of Figure 2a-b was designed in

this way. Figure 3a shows the gray-level histogram for the image in Figure 1, as well as the ranges used to produce the 8-level version in Figure 2. Gray level range and threshold selection can, of course, be done interactively. The effects of a poor choice of ranges is shown in Figure 3b.*

Display of small pieces of an image is usually adequate for checking results, and even for interacting by pointing or outlining, since the objects or regions to be outlined will usually be small relative to the entire image. At times, as in the examples in this paper, output of small pieces is all that is needed. If output of entire images is required, overstrike on the line printer can be used (remembering to override the page-skip!); a 500-by-500 point image can be output in four vertical strips, each a page wide and about six pages long. This requires much tearing and pasting, but has the merit of requiring no special equipment. If a microfilm output device with half-tone capability is available, it provides a far more compact output format. Alternatively, one can output images on tape, and take advantage of commercially available tape-to-image equipment.

---

*Figures 2a and 3b are best compared by viewing from a distance.

Interaction

The ability to execute an image processing operation and display the results already constitutes a high degree of interactive capability, since the user can determine what to do next in near-real time. The chief facility still lacking is the ability to point to objects in the image. It should be realized that in image processing, pointing can usually designate only a single image point, since the image has no underlying structural description that is known to the computer, as it has in computer graphics. In order to designate objects or regions in an image, one must outline these rather than simply point to them.

Region outlining can be done straightforwardly on either an alphanumeric CRT display or teletype printout of an image. In the CRT case,* the piece of picture in question is displayed by the computer in non-protected mode. The user switches the terminal to "batch mode" and uses the terminal cursor controls to position the cursor on the top row of the region to be outlined. At each point where the region boundary intersects the row, the user types a character (different from the one already typed). He then moves the cursor to the end of the row and transmits the "revised" line. The computer can now compare this to the original line and store the locations of the differences. When all rows containing region

*The exact procedure would vary somewhat, depending on the particular terminal used.

boundary points have been processed in this way, the computer can construct the complete region outline (and display it, if desired).

The procedure on a teletype is similar, but more tedious: the user informs the computer that he is about to input an image the same size as the displayed one. He then rolls the teletype paper back to the first line of the displayed image, and enters carriage returns until the top row of the region is reached. For rows that hit the region, the user spaces over to the region boundary points and overstrikes these points with an arbitrary character. Rolling the paper back simply establishes visual registration of the border and picture for the user; the computer is unaware of it, but can still register the two because they are the same size.

Both the CRT and teletype versions of the outlining procedure can be designed to permit correction of errors by re-inputting only the corrected border points, without having to re-outline the entire region. In any case, they are slow enough that errors are unlikely. The alphanumeric display functions as a half-tone which is coarse enough to permit precise outlining, yet provides sufficient grayscale to permit viewing the output as an image rather than as an array of discrete dots. The outlining procedure, it will be noted, requires no special interrupt priority; outlines are transmitted row by row exactly as in the ordinary use of the terminal.

A example of the use of outlining is given in Figure 4, which shows an outline in register with the picture of Figures 2-3. The display of the outlined region is done by a program which determines all points of the image that lie on or inside the outline. Once the outlined region has been extracted from its background, one can compute properties of the region, relating to its shape, texture, etc.

Software

The nature of the software used in an interactive image processing system will depend greatly on the machine being used, the programming talent available, and the types of processing operations to be performed.

Several major packages of image processing programs are available. Two notable examples are Vicar, developed by California Institute of Technology's Jet Propulsion Laboratory, and PAX, developed by the Universities of Illinois and Maryland. These packages are usually embedded in a high-level language such as Fortran. The examples given in this paper were programmed in PAX.

Historical Notes

Most of the major image processing research groups have developed interactive systems, but very few of these have been documented in the published literature. A highly nonrepresentative set of examples is [1-3]. On the use of overstruck characters to represent grayscale see [4].

PAX is a collection of over 100 basic image processing routines that can be called from Fortran programs [5]. Originally a simulator for the ILLIAC III computer, versions of PAX for several different machines are available, notably the IBM 7094 and 360/370 (50 and above); Univac 1108; CDC 3600 and 6600; and DEC-10.

References

1. I. H. Barkdoll and B. L. McGlamery, An on-line image processing system. Proc. ACM Natl. Conf., August 1968, 705-716.

2. L. Hodes, A programming system for the on-line analysis of biomedical images. Comm. ACM 13, May 1970, 279-283.

3. E. Arthurs, W. S. Bartlett, D. J. Ladd, R. L. Salmon, and J. H. Whipple, PICTURELAB--an interactive facility for experimentation in picture processing. Proc Spring Joint Computer Conf., May 1970, 267-273.

4. B. Perry and M. L. Mendelsohn, Picture generation with a standard line printer, Comm. ACM 7, May 1964, 311-313.

5. E. G. Johnston, The PAX user's manual, Computer Science Center, University of Maryland, June 1972.

Figure 1. 72-by-44-point, 32-gray-level image.

Figure 2. Use of alphanumeric arrays to represent images.

a) Image of Figure 1, with gray levels represented by characters as shown in (b).

Gray level(s)	Character
0-3	blank
4	.
5	-
6	+
7-8	I
9-11	A
12-16	*
17-31	W

Figure 2b. Characters used to represent gray levels in (a).

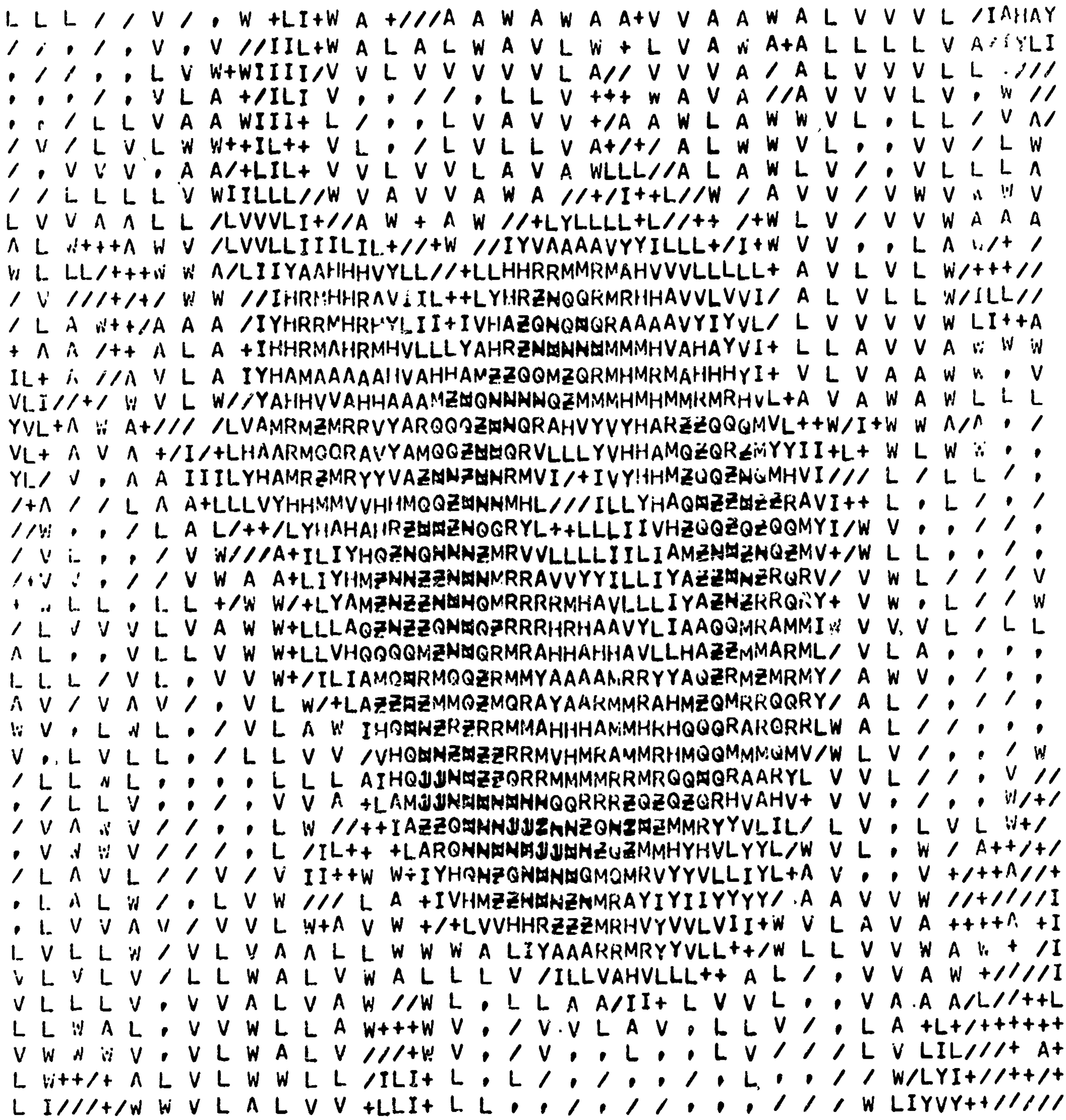


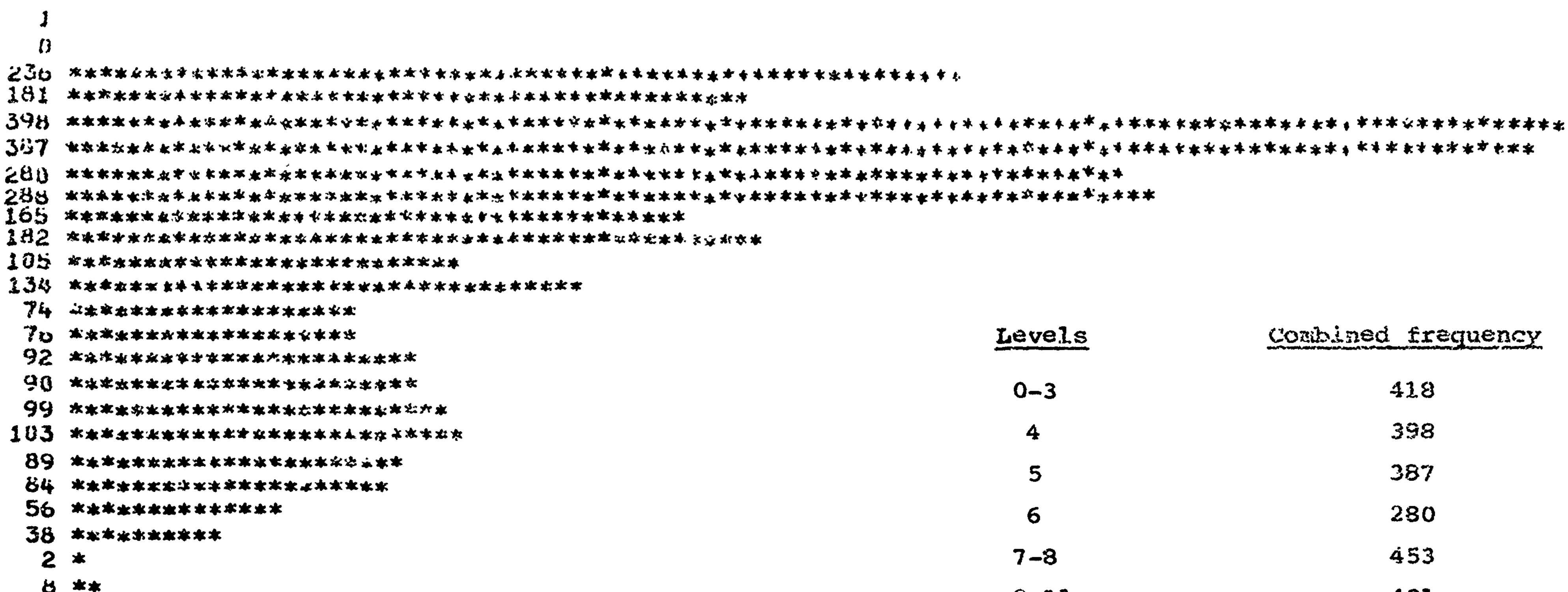
Figure 2c. Same image, with gray levels represented by pairs of overstruck characters as shown in (a).

Gray level	First character	Second character
0	- in odd columns	
1	. in odd columns	
2	, in odd columns	
3	/ in odd columns	
4	L in odd columns	
5	V in odd columns	
6	A in odd columns	
7	W in odd columns	
8	+	
9	/	
10	I	
11	L	
12	Y	
13	V	
14	A	
15	H	
16	R	
17	M	
18	Q	
19	Z	-
20	N	-
21	N	=
22	Z	=
23	J	1
24	W	/
25	D	/
26	D	1
27	N	1
28	M	1
29	N	Z
30	M	Q
31	W	$

Figure 2d. Pairs of characters used to represent gray levels in (c).

```
4544423232533226778BA877678999676777667667685555676676644454555469AEFEC
323223322552245699AAB87764476444766655477787455567766865444545455569ACBA
23333323234555787AAAA954554452545455554769965355566796644454545545478999
232222332354476689ABA5552522323223444456888676675466996454545445522476 99
2422334545546767 7AAA8545322322455566555789666775466677745544244545335669
3554344555545777 88AB8865442243245544445566898976 46777755452224555335 4477
322255545425666 98RAB8655554554554464546 67BBB99674666744553332556454 24767
33344445454456 7AABBB99755667555466776799 69A88B9976976555543554775776 7657
44555767674546 9BDDDBA899677686677799 8BCBBBB8B998869875455432555477676667
64457888677456 9BDDBBAAABAB89987799ACDEEEEDCCABBB89A877555523254466798697
7746BB988877776 9BAACEEFFFDCBB998BBFFGGHHGHEFDDDBBBB866665543544479888996
975799989897767799AFGHFFGEDAAB88BCFGJKIIGHGFFEDDBDDA9766445444467 9ABB995
954406788966666 79ACFGGHFGFCBAA8ADFEJIKILIGEEEEDCACDB9744555454557 6BA8865
87666798866546678AFFGHEFGHFDBBBCEFGJKLKKLHHHFDEFECDA864545675555677777 74
AB876699655544 67ACFEHEEEEEFDEFFEHJJIIHJIGHFHGHEFFFCA875445556767777 42554
DBA99897765546 799CEFFDDEFFEEEHJLIKKKKIJHHHFHFHHGHGFDB8675666776776444444
CDB865776899969BDEHGHJHGGDCEGIIIJLKIGEFDCDCFEGJJIIIHDB8879A8767769642232
DB866455678 9A98BFEEGHIIGEDCEHIIJLLIGDBBBCDFFEHIJIGJHCCAA8B8674467674 2222
CB9653226666AAABCFEHGJHGCCDEJLKJLKGHDA98ADCFFHJIIJKIHFDA99974435454 33222
98673332456668BBBDCFFHHDDFFHIIJLKKHFB999ABBCFEILJJLJJGEDA88645224 4332233
997422223547668 9889BCFEFEFGJLLJKIIGCB88BBBAADFJIIJIJIIHCA9775522233 23324
9752442322355779 9968ABACFIJKIKKKJHGDDBBBBAABAEHJKLJKIJHD8977454223223224
9855522233345776668BACFHJKKJJKLKHGGEDDCCABBACEJJLKJGIGD9557744233 23456
877544422245468976798BCEHJKJJKLKIHGGGGHFEDBBBACEJKJGGIGC8456762443323577
96445455524454667778BBBEIJKJJIKLIJGGGFGFEEDCBAEEIIHGEHHA745454544234 4545
664424235444425775 78BBEFIIIIHJKLIGHGEFFEFFEDBBFEJJHHEGHB945546652222 2325
474444345545225555789ABAEHILGHIIJGHHCEEEEHGGCCEIJGHJHGHC9664745323332222
67553354655233245447798BEJJLJHHIJHIGECEEGHHGEFHJIHGGIIGC976745322332 3324
775423467442223555446676AFILKJGJGGHHEFFFEHHFGFIIIGESIGGB756644333 232224
54244555444222354444555 69DFILKJLJJGGHDFHGEHHGFHIIHHHIHD9744454322223357 6
35454577442222222544454 76AFINNKLJJIGGHHHHGGHGIILIGEEGC475555423232235799
233445445422233222555466 8BEHNNKLLKLKKIIGGGJIJIJIGFDEFD87555423332225798 9
325565755232332223477699 8AAEJJILKKNNMKKJIKMLJHHGCCDBAB96445223445746 7896
24547474523332332346 9AB8868BEGIKKLKLNNLKJIJHHFCFDBCCB977554224779758898 9
35476555453232533256AA887678ACFIKJIKLKLIHIHGDCCDBBACB6665423245789886998
2346644475352345547799944 3678ADFHJJKLKJKHGECACAACCCC976665555577996 9999A
22445454645334545447786454778 98BDDFFGJJJHGFDCDDBDAA87755476455678 8878A
4554454574325544546767444577767767BACEEEGGHGCCDBB689744444545477677 6879A
54455544543547447767455676645444456 9ABBDEFDBBB88764433325555666768 9999A
54444444542255546444566677997442234545666 9AA874654554222225566676 9899888
44477765452354577547476678887452223355554667542544455233224466 8B8988888
547676775525554776654456999876522233552324442325455432333 34556BAB999876 6
46788987644452467776444 79ABA8745234432233522223325432222333679BCA899 8898
46A99989767455466745545 68BBA8645432323322333322232234323477BACDC8899999
```

Figure 2e. Same image, with gray levels 0,...,31
represented by blank, 1,...,9, A,...,V.



Levels	Combined frequency
0-3	418
4	398
5	387
6	280
7-8	453
9-11	421
12-16	431
17-31	380

Figure 3a. Gray level histogram for Figure 1; only levels 0,...,23 occur. Table shows ranges used in Figure 2a.

Figure 3b. Effect of a poor choice of gray level ranges: 0-3 = blank, 4-7 = ., 8-11 = -, 12-15 = +, 16-19 = I, 20-23 = A, 24-27 = *, 28-31 = W. Note that * and W are never used.

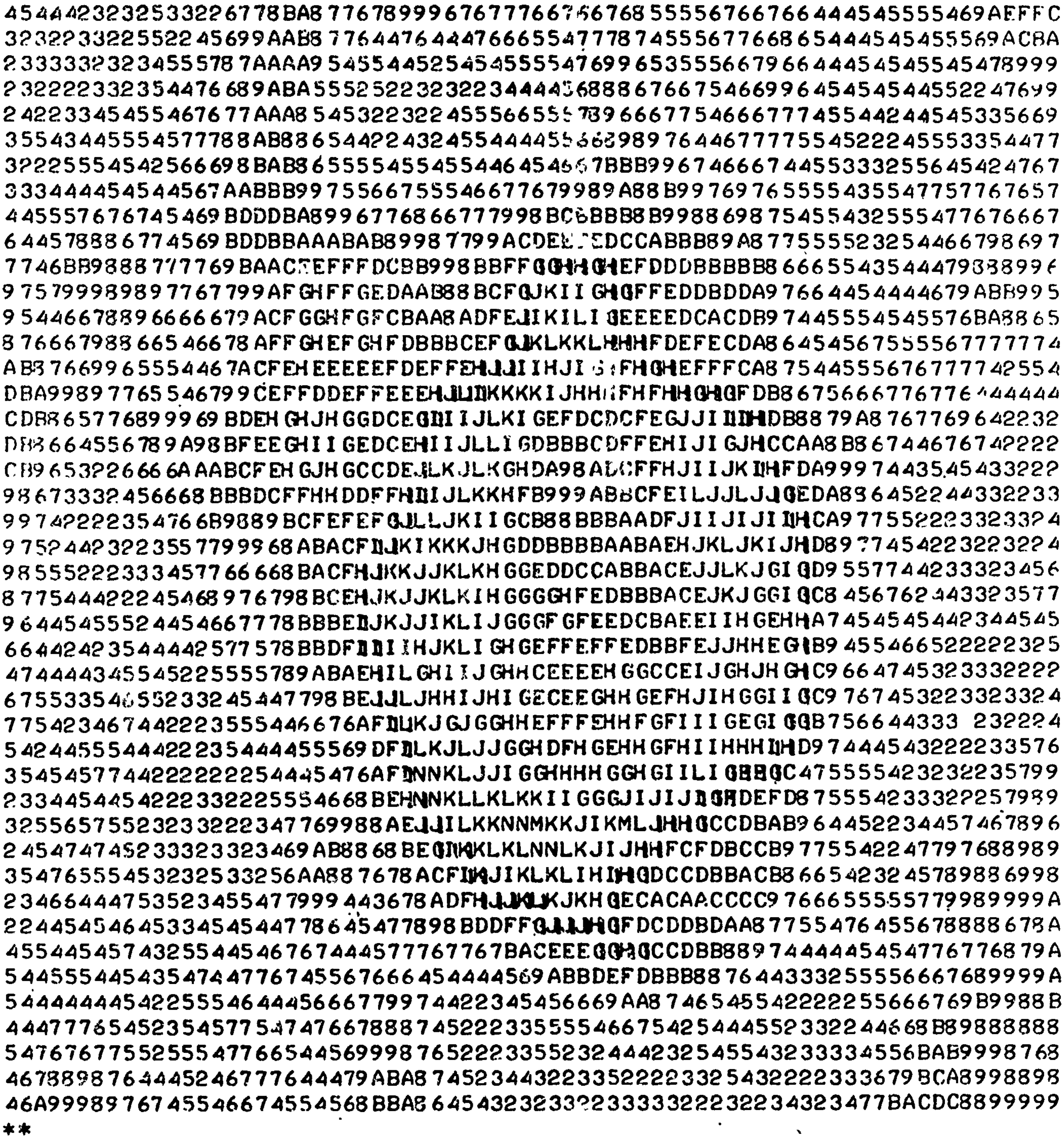


Figure 4a. Outline overstruck on a printout of Figure 2e.

```
            111111
           1     11
           1     1
          11     111
        111        111
     1111            11111
    11                   111
    1                      1
    1                      11
   11                       11
  11                         11
 11                           1
11                            1
1                             1
1                             1
11                            1
 1                            1
 11                           1
  11                         11
   1                        11
   1                     1111
   11                  111
    11              1111
     111           11
       11        111
        11111    1
            111111
                1111
```

Figure 4b. Printout of the outline alone, as a check.

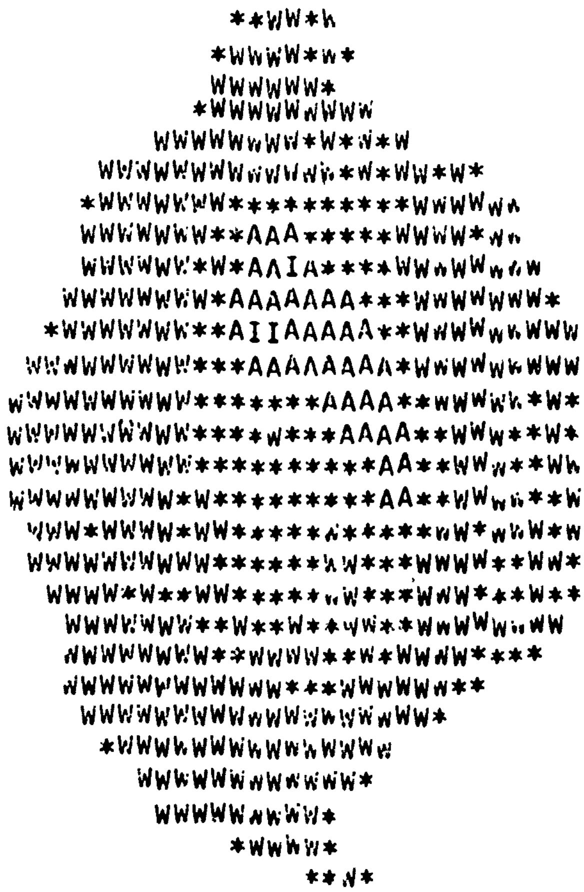


Figure 4c. Printout of the outlined region.

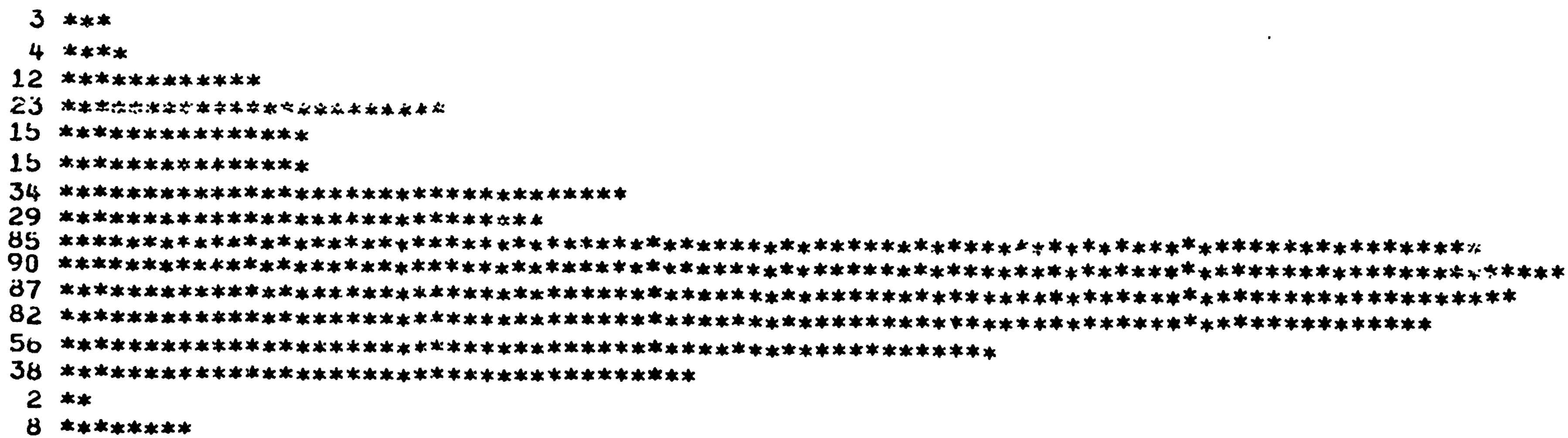


Figure 4d. Gray level histogram of the outlined region (horizontal scale four times that of Figure 3a). Only levels 8,...,23 occur. The region has area 583.